

Listing of Claims:

1. (Currently Amended)     A hardware-based multithreaded processor comprising:  
    a plurality of microengines, each of the microengines comprising:  
        a control store;  
        controller logic;  
        context event switching logic; and  
        an execution box data path including an arithmetic logic unit (ALU) and a general  
purpose register set, the ALU performing functions in response to instructions, one of the  
instructions causing the ALU to load ~~A computer instruction comprising: a instruction that loads~~  
one or more bytes of data within a transfer register associated with one of a the plurality of  
microengines with a shifted value of an operand that preserves the bytes of data that are not  
loaded.
2. (Currently Amended)     The processor ~~computer instruction~~ of claim 1 wherein the  
instruction further comprises ~~comprising:~~ a bit mask that specifies which of the one or more  
bytes of data are affected.
3. (Currently Amended)     The processor ~~computer instruction~~ of claim 2 wherein the bit  
mask indicates a left shift n bits, where n is a number from one to thirty-one.
4. (Currently Amended)     The processor ~~computer instruction~~ of claim 2 wherein the bit  
mask indicates a left shift by an amount specified in a lower five bits of the first operand of a  
previous instruction, where the lower five bits is a number from one to thirty-one.
5. (Currently Amended)     The processor ~~computer instruction~~ of claim 2 wherein the bit  
mask indicates a right shift n bits, where n is a number from one to thirty-one.
6. (Currently Amended)     The processor ~~computer instruction~~ of claim 2 wherein the bit  
mask indicates a right shift by an amount specified in a lower five bits of the first operand of a  
previous instruction, where the lower five bits is a number from one to thirty-one.
7. (Currently Amended)     The processor ~~computer instruction~~ of claim 2 wherein the bit  
mask indicates a left rotate n bits, where n is a number from one to thirty-one.

8. (Currently Amended) The processor computer instruction of claim 2 wherein the bit mask indicates a right rotate n bits, where n is a number from one to thirty-one.

9. (Currently Amended) The processor computer instruction of claim 1 wherein the instruction further comprises ~~comprising~~: an optional token that is set by a programmer and specifies to set arithmetic logic unit (ALU) condition codes based on the result.

10. (Original) A method of operating a processor comprising:

loading one or more bytes of data within a register associated with one of a plurality of microengines with a shifted value of an operand; and

clearing the bytes of data that are not loaded.

11. (Currently Amended) The method computer instruction of claim 10 further comprising: providing a bit mask that specifies which of the one or more bytes of data within the register are affected.

12. (Currently Amended) The method computer instruction of claim 11 wherein the bit mask indicates a left shift n bits, where n is a number from one to thirty-one.

13. (Currently Amended) The method computer instruction of claim 11 wherein the bit mask indicates a left shift by an amount specified in a lower five bits of the first operand of a previous instruction, where the lower five bits is a number from one to thirty-one.

14. (Currently Amended) The method computer instruction of claim 11 wherein the bit mask indicates a right shift n bits, where n is a number from one to thirty-one.

15. (Currently Amended) The method computer instruction of claim 11 wherein the bit mask indicates a right shift by an amount specified in a lower five bits of the first operand of a previous instruction, where the lower five bits is a number from one to thirty-one.

16. (Currently Amended) The method computer instruction of claim 11 wherein the bit mask indicates a left rotate n bits, where n is a number from one to thirty-one.

17. (Currently Amended) The method computer instruction of claim 11 wherein the bit mask indicates a right shift n bits, where n is a number from one to thirty-one.

18. (Currently Amended) The method computer instruction of claim 10 further comprising an optional token that is set by a programmer and specifies to load arithmetic logic unit (ALU) condition codes based on the result formed.

19. (Currently Amended) A method of operating a processor comprises:

loading one or more bytes of data within a register associated with one of a plurality of microengines ~~microengines~~ with a shifted value of an operand; and  
preserving the bytes of data that are not loaded.

20. (Original) The method of claim 19 further comprising:

providing a bit mask that specifies which of the one or more bytes of data within the register are affected.

21. (Original) The method of claim 20 wherein the bit mask indicates a left shift n bits, where n is a number from one to thirty-one.

22. (Original) The method of claim 20 wherein the bit mask indicates a left shift by an amount specified in a lower five bits of the first operand of a previous instruction, where the lower five bits is a number from one to thirty-one.

23. (Original) The method of claim 20 wherein the bit mask indicates a right shift n bits, where n is a number from one to thirty-one.

24. (Original) The method of claim 20 wherein the bit mask indicates a right shift by an amount specified in a lower five bits of the first operand of a previous instruction, where the lower five bits is a number from one to thirty-one.

25. (Original) The method of claim 20 wherein the bit mask indicates a left rotate n bits, where n is a number from one to thirty-one.

26. (Original) The method of claim 20 wherein the bit mask indicates a right shift n bits, where n is a number from one to thirty-one.

27. (Original) The method of claim 19 further comprising an optional token that is set by a programmer and specifies to load arithmetic logic unit (ALU) condition codes based on the result formed.